REMARKS

Claims 1 – 35, 44 and 45 are pending in the application.

Claim Rejections – 35 USC § 112

Claim 45 is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement. The Examiner argues that the term "Trading" which is used in the claim is not specified in the detailed description.

However, the term "Trading" is specified in the description (Summary of the invention), in lines 3-16, on page 14: " According to a further aspect of the present invention there is provided a method for facilitating interaction by a plurality of users at a plurality of client terminals with at least a first object, said first object having display and interaction characteristics and functional characteristics, in a networked virtual reality environment; said method comprising: encapsulating the display and interaction characteristics in a display and interaction part of said first object, encapsulating functional characteristics in a functional part of said first object, downloading said display and interaction part of said first object to user client terminals, and

retaining said functional part of said first object at a remote location networked with said user client terminals, said interactions comprising trading using said objects."

The meaning of "interaction comprising trading" as used in the summary of this invention should be obvious to anyone experienced in the art. There are many ways in which trading could take place using the present invention, via the kinds of relationships described in the application, thus, as an example, if one of the objects is a credit card and the other object is a chair and we place the chair on the credit card using any of the relationships described in the application, the computer may initiate electronic buying and the chair will be bought by the user.

Thus claim 45 is believed to be supported and should be allowable, as in the original application.

Claims 20-33 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject which applicant regards as the invention. More specifically it is argued by the examiner that these claims recite the limitation "virtual network computer environment" in line 1 of all the claims and that there is insufficient antecedent basis for this limitation in these claims. Appropriate correction has been made and claims 20-33 are now believed to be allowable in this respect.

Claim Rejections – 35 USC § 102

Claims 1-11,14-17,20-28,31-35,44 and 45 were rejected under 35 USC 102(e) as being anticipated by Sato et al. (US 6388667).

Favorable reconsideration of this rejection in view of the above amendments is respectfully requested since, as will be shown below, Sato fails to disclose a separately encapsulated virtual object in an object oriented environment. This object is managed as a single object having at least a user-sensible aspect and at least a functional aspect, the user-sensible aspect being encapsulated separately from the functional aspect.

The present application discloses a way to implement Object Oriented Programming (OOP) in multi-user, real time, virtual reality computer applications over networks while avoiding a significant problem associated with such prior art OOP applications, namely - the transferring of large amounts of data between networked computers which requires considerable network resources in terms of bandwidth and / or transfer time.

Object Oriented Programming (OOP) is a paradigm which breaks down problems into groups (classes) of related parts that take into account both the code and the data related to each group. The groups are then organized into a hierarchical structure and translated into subgroups called Objects.

The grouping of both code and data associated with a given object together constitutes one of the two key concepts in Object Oriented Programming, namely - Encapsulation, the other being Inheritance which allows us to capture the similarities and the differences between classes of objects.

With Encapsulation you can send an object, which as described above encapsulates all aspects of the object together, and it will work on another computer that does not have the identical program running. However the traditional OOP objects comprise all aspects associated with each object, including all object methods and data structures and are thus very "bulky" and produce very heavy transportation loads in a networked computers environment, particularly in data – intensive heavy applications like virtual reality applications.

This invention introduces a novel OOP encapsulated object which is also **split**, such that its user-sensible aspect is encapsulated separately from its functional aspect. With this splitting you can have parts of the object working on one computer and another part working on another computer, the object related data sent over the network is much smaller and the transportation load in a networked computer environment is significantly reduced.

Sato (US 6388667) discloses a totally different environment for the development of virtual reality applications than the one introduced in this invention. With the Sato environment, the actor is provided with both characteristics as objects and characteristics as processes (column 8, line14), however the Sato environment actor is **not a split** OOP encapsulated object, such that its user-sensible aspect is encapsulated separately from its functional aspect.

Claim 1 describes a virtual object for use in an object oriented environment; the virtual object comprising at least a user-sensible aspect and further comprising at least a functional aspect; the user-sensible aspect being encapsulated as a user-sensible encapsulation, separately from the functional aspect.

The examiner refers to Sato, particularly to elements 28 and 32 in Figure 2 as he argues that these elements illustrate such a virtual object which comprises at least a user-sensible aspect and further comprises at least a functional aspect, the user-sensible aspect being encapsulated as a user-sensible encapsulation, separately from the functional aspect.

However, these elements do not correspond to the same object. Actually element 32 in Figure 2 is an environment actor / object which relates to the control of the environment surrounding the actors: "The role of the environment actor is to control details such as the color of the background other than the stage 44, and the brightness of light sources " (Column 9, line 54) and actor 28 in FIG. 2 is responsible for the head of a character 42 on screen.

These actors do not constitute two separately encapsulated aspects of the **same** object but rather different objects, in which each object is encapsulated in the conventional way.

The importance of this is that in Sato, the same object cannot be run jointly on two different computers, and thus there is no advantage to be gained in saving bandwidth when communicating by two different computers over a network. By contrast with the split object of claim 1, it is possible to retain one part encapsulation and send the other part encapsulation and therefore use less bandwidth but have the object work substantially simultaneously on two computers.

Thus the features of claim 1 are neither shown nor hinted at by Sato and it is respectfully believed that claim 1 should be allowed.

Claim 2 explicitly recites the issue of partial communication of the split object over a network.

Claim 8 as amended describes a first virtual object within a virtual computing environment, the first virtual object having a relationship with a second virtual object, the relationship being such that an interaction with the first virtual object is operable to bring about a consequential interaction with at least a second object, the virtual computing environment comprising a method for restricting the number of consequential interactions of a virtual object with further virtual objects when a maximum number of consequential interactions is reached.

Sato fails to disclose the bringing about such a consequential interaction and a method for restricting the number of consequential interactions of a virtual object with further virtual objects when a maximum number of consequential interactions is reached.

Claim 15 describes a virtual reality environment comprising a scene and at least one virtual object supported by a scene database, the scene database having at least a first interchangeable functional unit associated therewith, the first interchangeable functional unit comprising functionality for the first virtual object, the virtual reality environment configured to support a method for facilitating interaction by a plurality of users at a plurality of client terminals with at least a first object, the first object having display and interaction characteristics and functional characteristics, the method comprising: encapsulating the display and interaction characteristics in a display part of the first object, encapsulating functional characteristics in a functional part of the first object, downloading the display part of the first object to user client terminals, and retaining the functional part of the first object at a remote location networked with the user client terminals.

Sato fails to introduce a virtual reality environment which comprises such a method comprising encapsulating the display and interaction characteristics in a display part of the one object, encapsulating functional characteristics in a functional part of the same object, downloading the display part of the first object to user client terminals, and retaining the functional part of the first object at a remote location networked with the user client terminals.

Actually, with Sato the object can be managed only in an environment which incorporates both aspects of each object and since Sato fails to introduce the encapsulation of a single whole OOP object, a whole object can not be sent over the network. Thus Sato Fig. 18c illustrates a host machine and terminals that are connected to it. If these terminals were fully enabled terminals having CPUs they might have taken the whole object had Sato introduced a single whole object encapsulation.

If these terminals are not fully enabled then the central machine processes the entire object. Either way Sato does not teach or even hint at splitting into two parts of a single object.

Claim 34 describes a dedicated control element for controlling the functionality of virtual objects belonging to a set of virtual objects within a virtual reality environment, the dedicated control element being associated with the virtual reality environment, and comprising: identification functionality for determining whether a virtual object within the virtual reality environment is a member of the set, and control functionality for processing events received from the identified virtual object, the control functionality is operable to bring about a consequential interaction of the virtual object with further virtual objects.

The environment introduced in this application comprises control functionality which may bring about a consequential interaction of a virtual object with further virtual objects, for example, if we move a table, the environment will automatically move with it all objects belonging to the set having an "on" relationship with the table.

Sato fails to disclose control functionality which is operable to bring about a consequential interaction of the virtual object with a set of further virtual objects, according to the identification of the set members.

Claim 35 describes a method for facilitating interaction by a plurality of users at a plurality of client terminals with at least a first object, the first object having display and interaction characteristics and functional characteristics, in a networked virtual reality environment; the method comprising:

encapsulating the display characteristics in a display and interaction part of the first object, encapsulating functional characteristics in a functional part of the first object, downloading the display and interaction part of the first object to user client terminals, and retaining the functional part of the first object at a remote location networked with the user client terminals.

The Examiner refers to Sato, particularly to elements 28 and 32 in Figure 2 and argues that these elements illustrate a virtual object which comprises at least a user-sensible aspect and further comprises at least a functional aspect, the user-sensible aspect being encapsulated as a user-sensible encapsulation, separately from the functional aspect. However, these elements are not *separate encapsulations* of different aspects of the *same* object. Actually element 32 is an environment actor / object which relates to the control of the environment surrounding the actors: "The role of the environment actor is to control details such as the color of the background other than the stage 44, and the brightness of light sources " (Column 9, line 54) and actor 28 in FIG. 2 is responsible for the head of a character 42 on screen, which is a part of the behavior of the character. These actors do not constitute two separately encapsulated aspects of the same object. Furthermore the head and like behavior actors are really part of the environment and may equally be applied to any of the characters.

Thus Sato fails to teach or even hint at separate encapsulation of visual and functional aspects of the single object and the downloading of one to a user computer combined with retention of the other at the originating computer. Claim 35 is thus respectfully believed to be allowable.

Claim 44 describes a method for controlling the functionality of a set of virtual objects within a virtual reality environment, comprising: incorporating allowable functionality for the set of virtual objects within a dedicated control element associated with the virtual reality environment, incorporating identification functionality within the dedicated control element to enable the dedicated control element to distinguish between virtual objects within the set and virtual objects not within that set, and thereby allowing the dedicated control element to control virtual objects within the set, the control element comprising a method for facilitating interaction by a plurality of users at a plurality of client terminals with at least a first object, the virtual object comprising at least a user-sensible aspect and further comprising at least a functional aspect; the user-sensible aspect being encapsulated as a user-sensible encapsulation, separately from the functional aspect.

Sato fails to teach or even hint at separate encapsulation of user-sensitive and functional aspects of the same single object and the downloading of one aspect to a user computer combined with retention of the other aspect at the originating computer.

Claim 45 describes a method for facilitating interaction by a plurality of users at a plurality of client terminals with at least a first object, the first object

having display characteristics and functional characteristics, in a networked virtual reality environment; the method comprising: encapsulating the display characteristics in a display and interaction part of the first object,

encapsulating functional characteristics in a functional part of the first object,

downloading the display and interaction part of the first object to user client terminals, and retaining the functional part of the first object at a remote location networked with said the client terminals, the interactions comprising trading using the objects.

As described above, Sato fails to disclose a method for facilitating interaction by a plurality of remote users with a single object by virtue of the object splitting between the different terminals.

Thus claim 45 should be allowed as in the original application.

The remaining claims mentioned in this section of the Office Action are believed to be allowable as being dependent on an allowable main claim. No new matter is added by the present amendments.

Claim Rejections – 35 USC § 103

Claims 12, 13, 18, 19, 29 and 30 were rejected under 35 USC 103(a) as being unpatentable over Sato (US 6388667) in view of Matsuda (US 6734885).

Favorable reconsideration of this rejection in view of the above amendments is respectfully requested since all the claims mentioned in this section of the Office Action are believed to be allowable as being dependent on an allowable main claim 8.

Furthermore as discussed above, Sato fails to teach a method restricting the number of consequential interactions between virtual objects. Matsuda does recognize "It is necessary to impose an upper limit on the number of clients allowed to participate in the 3-dimensional virtual space" (Column 2, line 31). However, Matsuda rather discloses an aura with a spherical shape having a predefined radius, formed around each object as a means to limit the sharing of information among objects BUT does not introduce a method for restricting the number of consequential interactions of a virtual object with further virtual objects when a maximum number of consequential interactions is reached, which is taught in the present disclosure.

Thus the combination of Sato and Matsuda fails to disclose a method for restricting the number consequential interactions of a virtual object to further virtual objects.

No new matter is added by the present amendments.

All of the matters raised by the Examiner have been dealt with and are believed to have been overcome. In view of the foregoing, it is respectfully submitted that all the claims now pending in the application are allowable.

An early Notice of Allowance is therefore respectfully requested.

Respectfully submitted,

Sol Sheinbein
Registration No. 25,457

Date: January 10, 2005